Increasing the Operational Value of Event Messages

Zhenping Li, Cetin Savkli

Lockheed Martin Space Operations
7500 Greenway Center, Suite 200
Greenbelt, MD 20770
Zhenping.Li@lmco.com

Dan Smith

NASA Goddard Space Flight Center (GSFC)
Code 581.0
Greenbelt, MD. 20771
Danford.S.Smith@nasa.gov

Abstract

Assessing the health of a space mission has traditionally been performed using telemetry analysis tools. Parameter values are compared to known operational limits and are plotted over various time periods. This presentation begins with the notion that there is an incredible amount of untapped information contained within the mission's event message logs. Through creative advancements in message handling tools, the event message logs can be used to better assess spacecraft and ground system status and to highlight and report on conditions not readily apparent when messages are evaluated one-at-a-time during a real-time pass. Work in this area is being funded as part of a larger NASA effort at the Goddard Space Flight Center to create component-based, middleware-based, standards-based general purpose ground system architecture referred to as GMSEC – the GSFC Mission Services Evolution Center. The new capabilities and operational concepts for event display, event data analyses and data mining are being developed by Lockheed Martin and the new subsystem has been named GREAT – the GMSEC Reusable Event Analysis Toolkit. Planned for use on existing and future missions, GREAT has the potential to increase operational efficiency in areas of problem detection and analysis, general status reporting, and real-time situational awareness.

1 Introduction

Extensive efforts have been directed at the efficient analysis and visualization of the satellite telemetry values over the past 25 years, but very little has been done to extract valuable information from the message logs generated within the mission control centers.

Traditional "event message" capabilities of both in-house and commercial mission control center systems are generally limited to filtering by time range and event type (telemetry, command, etc) for either real-time or off-line viewing. The recent efforts at NASA's Goddard Space Flight Center to develop a new standard-based ground system architecture provides the opportunity to greatly increase the operation values of the data contained in the event logs. As a part of the GMSEC control center architecture project, Lockheed Martin Space Operations is developing the GMSEC Reusable Event Analysis Toolkit (GREAT) to take advantage of the expanded event logs and standardized message definitions. GMSEC, with GREAT, is a new component based ground system architecture to utilize a mix of in-house and commercial products in support of current and future single and multi-satellite missions.

The new GMSEC architecture includes a standardized messaging service to allow subsystems to communicate with each other. The messages allow for communications management (publish/subscribe, message filtering, etc.) and for functional message content (data frames, parameter values, processing directives, and event messages). Event messages provide a standard means for any subsystem to report a particular activity or item of interest. Traditionally only generated by the real-time telemetry and command system, event messages are now also generated by the flight dynamics system and the planning and scheduling system and may even originate at the spacecraft or other ground system components. GMSEC's XML-based event message approach extends the standardized event message definition beyond the "time, type, and fixed length string text" format. New capabilities for event message display and operation concept could emerge from the considerably expanded event log. Data analysis and data mining are possible in addition to the simple monitoring. This could potentially provide significant benefits to the daily operations: the regular activity reports could be automatically generated from event logs, and the operational anomalies could be discovered quickly through the data analysis and data mining.. The focus of this paper is on the new capabilities and operation concepts being made possible as a result of the GREAT development effort.

This paper is organized as follows: Section 2 briefly discusses the GMSEC architecture and the standardized approach to the event messages. Section 3 discusses the general concept, capabilities, and design of GREAT system. Section 4 discusses the benefits of the new capabilities and operation concepts. The summary is given in Section 5.

2 System Level Architecture

The GMSEC architecture provides a flexible and cost-effective approach to meet the operational needs of the current and future missions over the next 5 to 10 years. The goal of GMSEC mission is to reduce the risk and cost of the development/integration and the continuing operations and to simplify the process of technology infusion over the life of a mission. Instead of selecting the "best in class" subsystem components, GMSEC strives for a plug-and-play capability so that a mission can select a major component for their system without having to replace the many other system components.

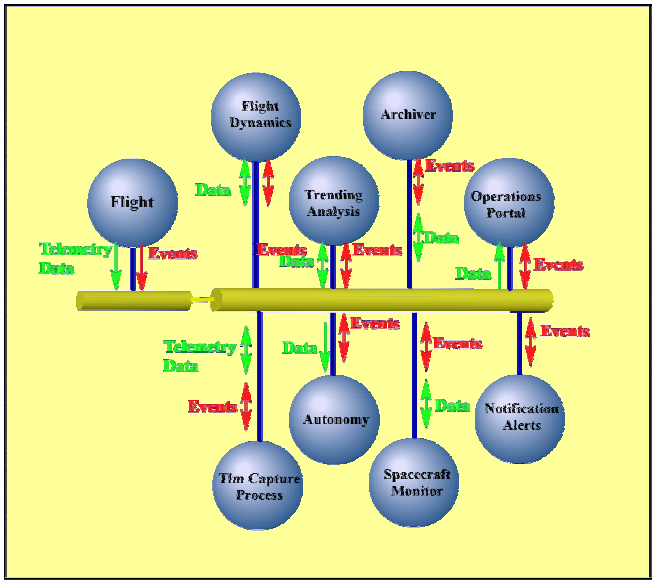

Figure 1 The GMSEC Architecture for spacecraft ground system

Figure 1 shows the main concept of the GMSEC architecture: it is component based with a centralized message service system. The message service system, sometimes referred to as a software bus or an information bus, is a middleware layer that itself is interchangeable with products such as the Java Messaging Service (JMS), or the Rendezvous and Smart Socket offerings from TIBCO[1]. Each subsystem, such as command and control, flight dynamics, or the scheduling and planning subsystem, is connected to the message service system with standardized interfaces and protocols. The event messages or data generated by each component are routed by the message service system to various destinations through the publish/subscribe or request/reply scheme. Key components that adhere to the GMSEC message standards have the "plug and play" capability and can be replaced by other components that better meet a mission's needs. The flexibility of the exchangeable component-based GMSEC architecture allows the creation of an integrated 'best of breed' ground and flight data systems based on the application of lessons learned and best practices, yet also allows missions to choose individual components without major impacts on the ground or flight system nor development/integration time.

The flexibility of the GMSEC architecture and the ability to add new components with a minimal integration effort is achieved by the standardization of the system interfaces. The GMSEC standardization efforts are two-fold: an open standard API for the programming interface between the component and message service system that allows the point-to-point and multi-cast communications with certain levels of the quality-of-service, and the standard schema for event message, telemetry, directive, data value, and data transfer [2]. The message schema is defined using Web Service Definition Language (WSDL). Key message definitions have been drafted and are being prototyped and validated prior to formal release.

The main entries in GMSEC event message schema are shown in Figure 2[3]. The system event message schema is extended beyond the "time, type, fixed length text string" format, and the event message schema consists of the header, source, and the content sections. In addition to the expanded definitions of the message header and sources, the message content section also has an entry for the detailed information, much like the body text of an e-mail message. The detailed information in the event message is not a fixed length string, and not limited to the single lines. Thus, a new approach to the event message display is warranted, as the traditional approach does not provide the platform to display the detailed information entry that is not limited to a single line.

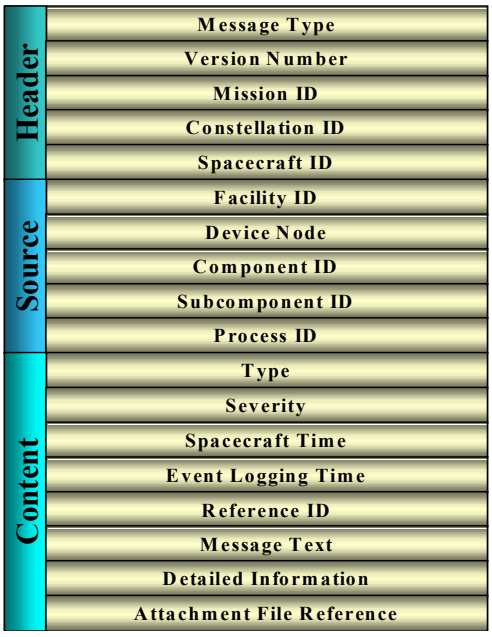

Figure 2 The GMSEC Event Message Schema consists of the header, source, and content sections.

Every component in GMSEC architecture publishes event messages for the real time monitoring and archiving so that every component leaves the trace or record of its activities in the event logs. For example, the orbit event files will be converted into a series event messages, the command load data can be converted into the stored command execution events, and the schedule can be translated into a list of planned activities. The expanded event log and definition provides a very broad context to analyze the system performance and potential problems. The consolidated event log from multiple systems can also increase the consistency within the control center and allow for centralized management and operations. This is particularly useful for missions with a constellation of spacecraft.

3 Event Analyzing System Concepts And Design

The GMSEC Event Analysis Toolkit, GREAT, provides a comprehensive system for the real time event display, event archive, historical event retrieval, event triggered actions, bulk event file handling, and automated report generation from the event statistics collection, event data analysis and data mining. It is a "plug and play" component under GMSEC architecture, and is also capable of working with the legacy system through the "plug and play" network interface and configurable message schema. GREAT separates the generic features from the mission specific setup features to make it more portable. The mission specific features, such as the message schema, message filtering scheme, and the user's preferences, are stored in an XML configuration file so that it can be reconfigured easily for different missions. The common interfaces are also defined so that the customized routine for the network interface, data processing, and event-triggered actions can be integrated with GREAT through 'plug and play'. The latest JAVA and J2EE technologies are being used in the development, which not only provides the flexibility to run on many platforms, but also reduces the development cycle significantly with the many open source tools and libraries.

Figure 3 shows the high level GREAT architecture. It has simple client-server architecture, in which the GREAT subsystems work as the clients for the COTS data server and the message service system under GMSEC architecture. The database server stores the event log database and other relevant databases. JDBC and SQL technologies, widely used in E-Commerce applications, are implemented for the interfaces between the clients and the database server. GREAT components receive the system event messages from the message service system under GMSEC architecture through the publish/subscribe scheme. The following sections discuss the six key subcomponents of the GREAT system.

3.1 Event Data Archive

The event log database is created using the GMSEC event message schema. Event Data Archive subsystem subscribes to every event message from the message service system, and ingests it into the database in the real-time. The archive will be used latter for the historical event retrieval and report generation. The event archive also provides data management capabilities, such as the creation of new event tables over fixed periods.

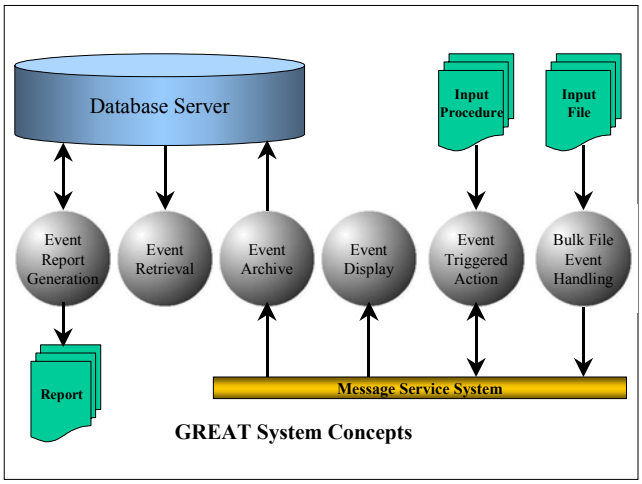

Figure 3 GREAT Architecture utilizing the publish/subscribe scheme to receive the event messages from the message service system, and performing the data archive, historical event retrieval, report generation from the database server.

3.2 Event Display

The event display software displays the filtered event messages in the real-time. Figure 4 shows the front panel of GREAT event display subsystem. The event display implements an 'e-mail' style to accommodate the expanded event message definition under GMSEC architecture; the detailed information defined in the message content is regarded as the message body, and the rest of the sub-fields in the header, source, and content sections are the header information of an 'e-mail' that are displayed in the main window. The complete message that includes the detailed information is displayed in a separated window if the user clicks the corresponding message in the real time window, similar to the standard e-mail display software. The additional information displayed on the separate window is no-longer limited to single lines. The detailed information for a command event could include an info block showing the raw uplink values, and for an expert system event could include the recommendations for corrective actions or the explanation of rule violated.

Because of the expanded event message definition, the message filtering is no-longer limited to the message-type. GREAT implements a generic filtering scheme that allows any sub-field with fixed values to be filtered. The filtering

scheme is contained in the XML configuration file, and can be configured with a GUI interface.

In addition to the flexibilities of various fonts and sizes on the real-time display, the GREAT real time display system also provides the capabilities to reorganize the messages being displayed. Any data column can be sorted by clicking the corresponding header on the display, which is not limited to the time tag or the message. The search functionality allows the pattern search of the message text in addition to the simple text search, and the search results are displayed in separated windows to allow users to save, copy, or print.

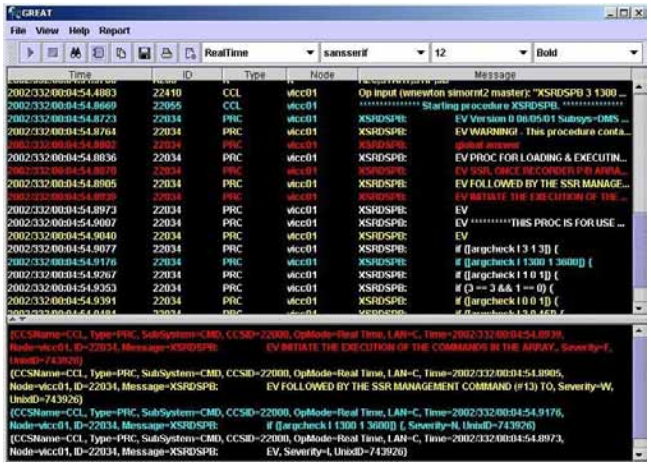

Figure 4 GREAT Real Time Event Display, the bottom panel displays 'e-mail' style full message.

3.3 Event Retrieval

The event retrieval subsystem uses SQL query to select event messages from the event database. It is invoked from the real-time display panel shown in Figure 4. The historical archive data retrieval goes beyond the practice of retrieving a single block of the data within certain time periods. The data retrieval panel provides users the option to enter the constraint implemented in the query language in addition to the time period. Therefore, retrieving historical event message becomes more efficient and flexible with only the data needed. The pattern matching is allowed for any column in database; this provides powerful search capability in the historical databases. In addition, the historical event display implements the tabbed panel, which allows multiple retrievals to be displayed at the same time. One can also perform searches on the historical data that has already been retrieved – allowing progressive refinement of a detailed search.

3.4 Event Report Generation

GREAT report generation subsystem provides a platform for statistical collections, data analyses, and data mining of the event databases. Figure 5 shows the high level architecture of the report generation subsystem. It contains two main components that interface with the database server: the report generation module and the data processing module. The report generation module is an open-source reporting tool, Data Vision, which is developed and maintained by Jim Menard[4]. It interfaces with the database server, and uses SQL query to extract the relevant information. The output report is generated in the form of HTML, PDF, XML, LaTex, or simple text format files. Data Vision has a GUI front that presents the full view of the database, user designed columns, and customized formula. The report can be designed and edited with GUI front using its "drag and drop" capability. The report template from the design panel is saved in an XML file. The report generation can also be executed on the command line if the report template already exists. In addition to SQL query, the extract data can be manipulated with user defined formula using the scripting language JRuby[5]. The virtual table linking capability in Data Vision could combine different data tables to perform more complex query.

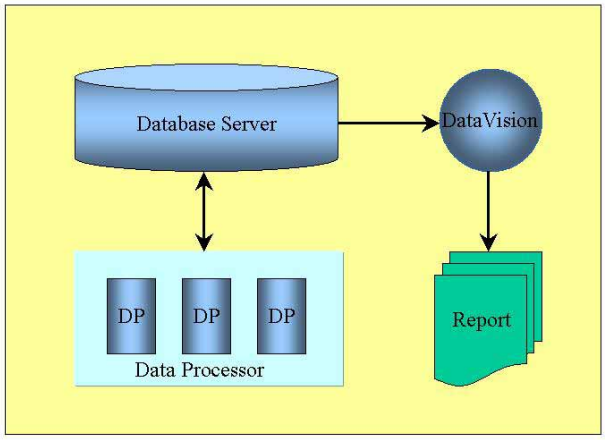

Figure 5 Report Generation subsystem contains the report generation and the data processing modules that interfaces with the database server.

Data Vision is integrated with GREAT for report generation from the event database retrievals covering a specified time period, which can be invoked from the GREAT panel in Figure 3. It provides a basic capability for extracting the needed data from database and generating the corresponding reports for a given period. More complex reports require additional data manipulating capabilities, and in some case, more advanced algorithms. GREAT provides a platform to make this possible with the data processing module. The data processing module is in fact a "container" for the data processing component with a defined common interface. The customized class for the mission specific task that implements the data-processing interface is invoked by the container. The data processing module uses SQL query to select the relevant inputs from

the system event database, and outputs the results in the database table form inserted back to the data server. The output table from the data processing component will be used as input by Data Vision to generate the relevant report. Generally, the data processing component provides an algorithm for extracting the data from the system event database to perform analyses or data mining, the output could be the high level summery of data, or the correlation among the various components in GMSEC architecture.

By providing the consolidated system event database from every component under GMSEC architecture, such as the real time command and control, flight dynamics, telemetry, and the planning and scheduling, GREAT report generation subsystem becomes an ideal tool to generate the periodic report on daily activities, and to study the correlation between various subsystems. The possible reports for a control center from the statistical collection, data analyses, and data mining include the PrePass and PostPass summary reports, end of shift report, daily report, and monthly reports, as well as other customized reports, which could provide the information on data integrity, command activity, or contact activity. These reports could provide the performance metrics for evaluating the quality of service and identify the operational trends and system anomalies for the spacecraft or ground system.

One common feature in the regular reports from various missions is the statistical collections between the pairs of the events, such as the AOS/LOS, out-of-limits/in-limits, and sign-on/sign-off events contained in the system event database. The activities in the regular reports include the command activities during AOS/LOS period. With the GREAT report generator, one could also provide the statistical information on

- How many minutes of contact time being scheduled during the week
- How many actual minutes of the contact during the week
- How many times, and for how long, was bus-current low
- How many commands uplinks are performed per day or per pass

Since the times of the event pairs are located in separate events, the direct implementation from Data Vision using SQL query would be very difficult. To implement the reports on the events during the pairs of events, a data-processing component that creates a generic data table is developed for the pair of events. The component selects the events with the event pair signatures, and groups them together to create a new database table that contains the times of the event pairs, such as the AOS/LOS time, as well as other relevant information, such as the duration between AOS/LOS times. The new database table is then inserted back into the database after the data is processed, which can be accessed by Data Vision. The Data Vision report tool can collect the statistical information on the minutes of the contact time from the new data table. It could also link the new database table with the event message table, and use the times of the event pair in the new table as the constraints to query the system event data table, and select any information during the periods of AOS/LOS, such as the information on command activities, data quality, or the telemetry. This implementation could meet the requirements of the regular report generations for many missions.

Further implementations of advanced data mining algorithms to study the correlations among components are planned. The generic decision tree algorithm[6], and neural network algorithms[7] are being developed that could be implemented as the data processing components.

3.5 Event Triggered Actions

The event triggered actions subsystem is activated when a pre-defined event message is received. The core can be viewed as a criteria/action table developed by the user over time. It provides a useful automated tool to the operations. The examples of the pre-defined actions include:

- Selective notification to relevant personal of changes of spacecraft or system status through e-mail or paging if the system receives a high severity event message.
- Automated generation of a summary report or summary of event messages through the report generation subsystem. For example, the post-pass report could be automatically generated after the LOS event message.
- Activation of a procedure or individual system directive.

The event triggered action subsystem is integrated with the event archive subsystem to monitor the unfiltered event messages. The pre-defined event messages are written into an XML input file along with the names of the action classes to be activated. The action classes are defined with a common interface so that they can be invoked by the event triggered action subsystem. Therefore, other customized event-triggered actions could also be implemented in this approach. The action classes for the selective notification through e-mail or paging and the automated report generation will be developed in the near future.

3.6 Bulk Event File Handling

Most of the products from the scheduling and planning system and the flight dynamics systems are formatted text files or are easily converted into time-tagged events. The bulk event file handler reads these files and generates the corresponding events messages so that the events can be archived into the database. A key aspect of this subcomponent is file version management – since bulk files

often cover extended time periods and are often updated prior to the end time of the original file.

4 Operational Value

Collectively, the six subcomponents of GREAT provide powerful operations capability not available in today's ground systems at GSFC.

The inclusion of event messages covering all aspects of mission operations provides a more complete view for mission monitoring and a new context for problem analysis. Expert systems and even small add-on tools in the GMSEC architecture use event messages as a primary mechanism for communicating status changes and identifying activities. Using the bulk file feature, mission schedules and orbital information can be converted to individual time-tagged event messages.

A single operator can now view the operating status of all mission control components and be immediately aware of problems with orbit computations, long-term trending, etc. and also be made aware of normal status changes such as load generation completion or schedule inputs received from the science team.

When an event message shows a telemetry value is out of limits, one past approach for analysis has been to plot the parameter's value around the time of the "anomaly". By looking at the expanded events log, the engineer may now see that the spacecraft had entered an eclipse period five minutes earlier or that a stored command loaded earlier that day just been executed.

With a wider breadth of event message types and some basic data mining capabilities, new types of problems can be identified and some routine reports can be automated. Seemingly random data dropout messages can be analyzed and possible correlations to a specific remote ground station configuration identified. Problems need not be limited to just spacecraft subsystems. Daily support logs showing the passes supported and the commanding activities and alarm activity for each can easily be built.

Being able to trigger activities based on event message text or groupings allows for immediate operator notification and allows for many functions to be automated. Upon detection of specific events, operators can be paged or e-mails can be sent to engineers. By allowing the same criteria to trigger the generation of user directives, corrective action can be taken, plots can be generated when key parameter limits are exceeded, etc. Over time, the list of criteria and resulting actions can expand to handle specific anomalies.

The GREAT system is being built to allow optional body text and attachments. As subsystems evolve to take advantage of these capabilities, a number of powerful scenarios will be possible. An expert system, upon detecting a subtle spacecraft problem, could issue an event message with the basic text being a notification of the problem. The body text could show the details of the problem and the actual rules violated. Files could be attached, one showing the spacecraft configuration at the time of the anomaly and another containing a recommended script for corrective action. Another use would be to automatically generate shift change reports, including summaries of each pass's activities and any problems recently noted.

5 Summary

The expanded event logs and message definition under the GMSEC architecture provide opportunities for new approaches and operation concepts to manage the system events, which will lead to more automation and autonomy for the spacecraft control. It provides an ideal ground to perform the data analyses and data mining to investigate the system performance and identify operation trends and system anomalies. These new approaches and operational concepts could potentially provide significant benefits to the operations teams.

Certainly more capabilities and operation concepts will emerge in the future beyond the contents being discussed here. Capabilities of today's e-mail systems will continue to be evaluated and some may be added to the GREAT design. Imagine the spacecraft and the ground communicating using e-mail, with loads and dumps as attachments, responses automatically generated, important messages sorted to different in-boxes and users working on actions on a prioritized basis. The Lockheed Martin GMSEC Reusable Events Analysis Tool represents a significant improvement over today's events subsystems and lays the groundwork for this extended future functionality.

Acknowledgement

Authors acknowledge the contributions from Jim Fessler, and Luanne Spiller of Lockheed Martin Space Operations.

Reference

[1.] See www.tibco.com for detail information on Rendezvous and Smart Socket.

[2.] Smith Dan, "Increasing The Operational Value of Ground System Event Messages", Ground System Architecture Workshop 2003, Manhattan Beach, California.

[3.] GMSEC Message Working Group, "GSFC Mission Service Evolution Center (GSFC) Interface Specification Document", April, 2003

[4.] See datavision.sourceforge.net for detail information on Data Vision.

[5.] See www.ruby-lang.org/en/ for detail information on JRuby.

[6.] Russell, S., and Norvig, P., "Artificial Intelligence", Prentice Hall, 1995, and reference therein.

[7.] Feraud, R., and Clerot, F. "A Methodology To Explain Neural Network Classification", Neural Networks, Vol. 15, pp. 237-246, (2002).